What is claimed is:

1.   A method of improving data transfer in a computing network, comprising steps of:

    receiving one or more packets to be routed to or from a plurality of virtual servers;

    providing an internal routing table for data link layer routing, wherein entries in the internal routing table are learned dynamically while processing the received packets; and

    using the internal routing table for routing the received packets.

2.   A method of improving data transfer in a communications network, the method comprising steps of:

    providing a concentrator that combines traffic from a plurality of virtual servers into a single outbound stream; and

    routing packets of the combined traffic, further comprising steps of:

        intercepting packets of the traffic at a data link layer of a communications protocol stack;

        comparing a destination address of each intercepted packet to entries in a data link layer routing table;

        forwarding the intercepted packet to a higher layer of the communications protocol stack if no matching entry is found by the comparing step, for routing by the higher layer; and

        performing data link layer routing of the intercepted packet, without intervention of the higher layer, if a matching entry is found by the comparing step.

1   3.   The method according to Claim 2, wherein the step of performing data link layer routing

2   further comprises steps of:

3   replacing the inbound packet header of the intercepted packet with an outbound packet

4   header using information from the matching entry, thereby creating a modified packet header; and

5   forwarding the intercepted packet using the modified packet header.

1   4.   The method according to Claim 2, wherein the entries in the data link layer routing table

2   are dynamically learned.

5.   The method according to Claim 2, wherein one or more of the virtual servers are

application servers.

6.   The method according to Claim 2, wherein the virtual servers each operate in a logical

partition within a single computing device.

1   7.   The method according to Claim 2, further comprising the step of deleting selected entries

2   from the data link layer routing table when the selected entries become obsolete.

1   8.   A system for improving data transfer in a communications network, comprising:

2   means for providing a concentrator that combines traffic from a plurality of virtual servers

3   into a single outbound stream; and

4   means for routing packets of the combined traffic, further comprising:

5 means for intercepting packets of the traffic at a data link layer of a

6 communications protocol stack;

7 means for comparing a destination address of each intercepted packet to entries in

8 a data link layer routing table;

9 means for forwarding the intercepted packet to a higher layer of the

10 communications protocol stack if no matching entry is found by the means for comparing, for

11 routing by the higher layer; and

12 means for performing data link layer routing of the intercepted packet, without

13 intervention of the higher layer, if a matching entry is found by the means for comparing.

9. The system according to Claim 8, wherein the means for performing data link layer

routing further comprises:

 means for replacing the inbound packet header of the intercepted packet with an outbound

packet header using information from the matching entry, thereby creating a modified packet

header; and

6 means for forwarding the intercepted packet using the modified packet header.

1 10. The system according to Claim 8, wherein the entries in the data link layer routing table

2 are dynamically learned.

1 11. The system according to Claim 8, wherein one or more of the virtual servers are

2 application servers.

1   12.   A computer program product for improving data transfer in a communications network,

2   the computer program product embodied on one or more computer readable media and

3   comprising:

4   computer readable program code means for providing a concentrator that combines traffic

5   from a plurality of virtual servers into a single outbound stream; and

6   computer readable program code means for routing packets of the combined traffic,

7   further comprising:

8   computer readable program code means for intercepting packets of the traffic at a

9   data link layer of a communications protocol stack;

10   computer readable program code means for comparing a destination address of

11   each intercepted packet to entries in a data link layer routing table;

12   computer readable program code means for forwarding the intercepted packet to a

13   higher layer of the communications protocol stack if no matching entry is found by the computer

14   readable program code means for comparing, for routing by the higher layer; and

15   computer readable program code means for performing data link layer routing of

16   the intercepted packet, without intervention of the higher layer, if a matching entry is found by the

17   computer readable program code means for comparing.

1   13.   The computer program product according to Claim 12, wherein the computer readable

2   program code means for performing data link layer routing further comprises:

3   computer readable program code means for replacing the inbound packet header of the

4 intercepted packet with an outbound packet header using information from the matching entry,

5 thereby creating a modified packet header; and

6 computer readable program code means for forwarding the intercepted packet using the

7 modified packet header.

1 14. The computer program product according to Claim 12, wherein the entries in the data link

2 layer routing table are dynamically learned.

1 15. The computer program product according to Claim 12, wherein one or more of the virtual

2 servers are application servers.